上周主要工作进展

1. 汇编代码完成resnet conv 1\*1的部分，尚未上板调试；
2. 阅读4篇图卷积相关文献，文献小总结见后；
3. 运行了图卷积模型的软代码实验，其突出特点是权值分布比较集中利于量化、网络稀疏度较高且稀疏度分布不均匀；

下周工作计划：

1. 完成resnet汇编的上板调试工作；
2. 针对目标GCN模型稀疏度较高的工作，阅读剪枝及稀疏矩阵加速的相关论文；
3. 绘制目标GCN模型的数据流图以评估其在MT上的可行性；

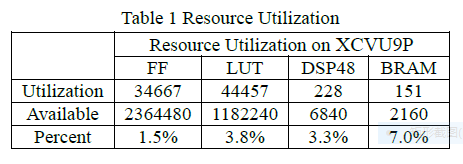
文献总结

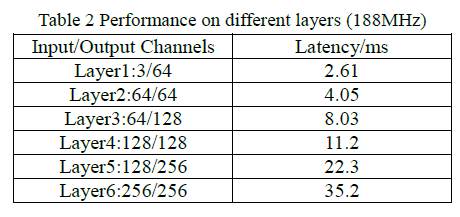
Accelerating Large Scale GCN Inference on FPGA, 2020 FCCM

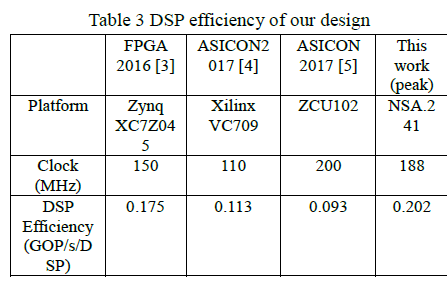
1. 采用了传统的压缩矩阵存储格式进行计算，加速器设计针对含图操作的图卷积网络；
2. 使用了节点重排序方法 (reordering)，消除图中度数较大的节点，其目的是使得每个PE都达到负载均衡的状态；
3. 在FPGA上实现，达到了相对CPU 30x和相对GPU 2x的加速比。

An FPGA Implementation of GCN with Sparse Adjacency Matrix, 2019 IEEE International Conference on ASIC

1. 加速对象是ST-GCN，一种不含图操作（如搜索节点等）的图卷积姿态识别网络。ST-GCN是学术界第一种使用图卷积的姿态识别方法，相比CV方法发姿态识别取得了SOTA的成果；
2. 文章使用了量化+传统稀疏矩阵乘的方法；
3. 在FPGA上实现，延迟和GPU相当的同时取得了相对GPU 6x的吞吐率提高；



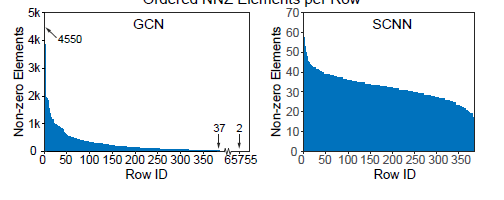


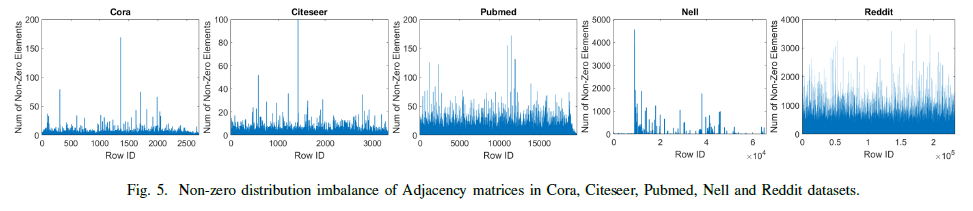


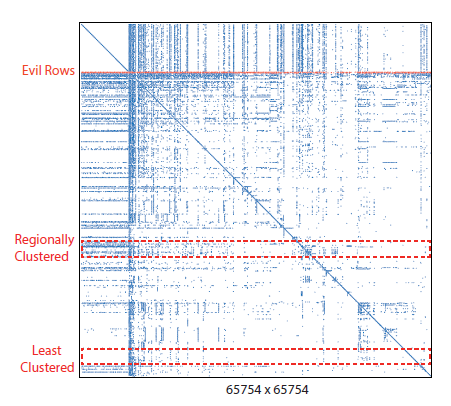
1. 本文相对于其他FPGA设计，更关注DSP的利用率。

AWB-GCN: A Graph Convolutional Network Accelerator with Runtime Workload Rebalancing, to be appeared on 2020 HPCA

1. 文章面向通用图卷积模型（含图操作的和不含图操作的），主要的着眼点在于大图计算中，图数据结构存在稀疏度极高(>99%)和非零元素分布不均匀的特点。文章阐述并分析了图数据结构的特点：80%的非零元素分布在20%甚至更少的行和列上；

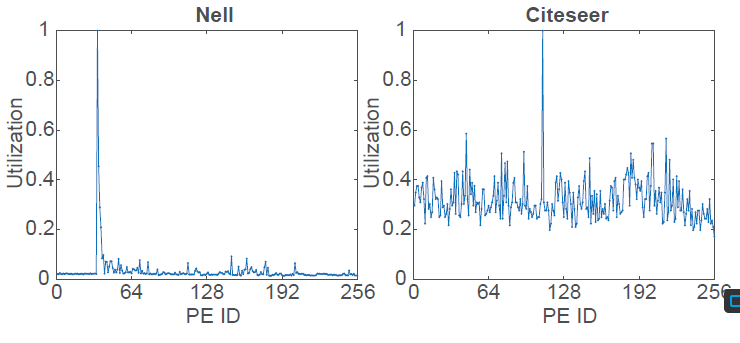




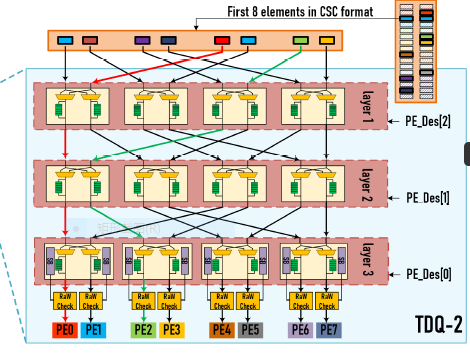


（蓝色的点线为矩阵中的非零元素）

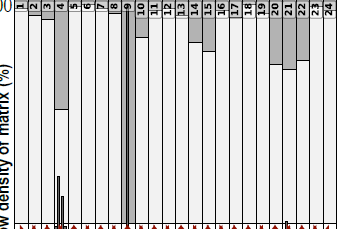
1. 文章提出了一种基于传统压缩矩阵格式的PE及其阵列组合作为baseline。文章证明了在没有额外优化的情况下，图数据分布的2-8定律会使PE负载及其不均衡，导致资源利用率不高和拖慢整体计算速度。



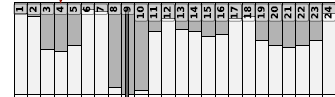
1. 针对（2）中的现象，文章提出了负载均衡的方法来提高PE利用率并缩短整体计算时间。文章将（2）中的PE进行重新组合，相邻的n个为一组。文章设计了一种任务队列硬件结构，该队列结构可以负责维护组内各PE的负载均衡，将等待计算的任务（在图卷积计算中通常对应矩阵的某个位置或某个基本分块）平均的分发给组内的各个PE；

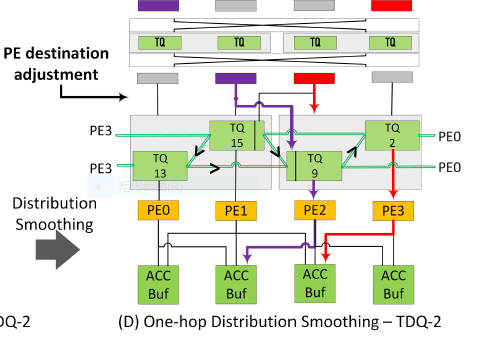


1. （3）中的方法针对的是基本矩阵分块内部的负载均衡，但（1）（2）中的图例表明，图卷积矩阵在更大的范围上存在数据分布不均衡的问题，对应硬件则表现为某些PE组相对空闲，但某些PE组极其繁忙。



文章在每个PE组内部都加入了1-hop的相连通信网络，每个PE组都可以将自己需要等待的任务分发给1跳内空闲的邻居，若邻居的负载也很高，则继续向下一跳分发。如果分发范围由1-hop扩展到n-hop，则runtime均衡效果会更好，但n-hop的设计会带来更多的硬件资源开销，因此文章最终还是采用了1-hop的设计。





（5）文章使用了ASIC平台进行验证，在4096PE的设计下取得了相对于CPU2622x和相对于GPU136x的加速比。